Reduced Resistive Magnetohydrodynamics with Implicit Adaptive Mesh Refinement[1]

Bobby Philip[2] Michael Pernice[2] Luis Chacón[3]
Los Alamos National Laboratory
Los Alamos, NM 87545

SIAM Annual Conference 2006
July 10–14, 2006

---

[1]This work was performed under the auspices of the National Nuclear Security Administration of the U.S. Department of Energy by Los Alamos National Laboratory, operated by Los Alamos National Security LLC under contract DE-AC52-06NA25396. Los Alamos National Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness. LA-UR 06-4521.

[2]Computer, Computational, and Statistical Sciences Division

[3]Theoretical Division

Outline

- Motivation
- Current-Vorticity Formulation
- Physics-based Preconditioner
- Numerical Results
  - Tearing mode
  - Island coalescence
  - Tilt instability
- Conclusions

Motivation

Models of resistive MHD contain multiple length and time scales.

- Local refinement in space can add resolution only where needed.
- Implicit time integration can more efficiently resolve the time scales of interest.
  - Explicit methods require $\Delta t \lesssim \mathcal{O}(\Delta x^2)$ when diffusion/Hall effects are significant.
  - Semi-implicit methods allow $\Delta t \lesssim \mathcal{O}(\Delta x)$.
    - Accuracy often requires somewhat smaller values.
    - Stability under long-term integration can be a problem.
  - Implicit time steps are constrained only by accuracy requirements.

Current-Vorticity Formulation of Reduced Resistive MHD[4]

$$\begin{aligned}(\partial_t + \mathbf{u}\cdot\nabla - \eta\Delta)\, J + \Delta E_0 &= \mathbf{B}\cdot\nabla\omega + \{\Phi,\Psi\} \\ (\partial_t + \mathbf{u}\cdot\nabla - \nu\Delta)\,\omega + S_\omega &= \mathbf{B}\cdot\nabla J \\ \Delta\Phi &= \omega \\ \Delta\Psi &= J\end{aligned}$$

on a rectangular domain Ω. Here,

$$\mathbf{u} = \nabla\times\Phi, \qquad \mathbf{B} = \nabla\times\Psi, \qquad \{\Phi,\Psi\} = 2[\Phi_{xy}(\Psi_{xx}-\Psi_{yy}) - \Psi_{xy}(\Phi_{xx}-\Phi_{yy})].$$

Equilibrium sources are chosen to balance prescribed initial conditions:

$$E_0 = \eta\Delta\Psi_0, \qquad S_\omega = \nu\Delta\omega_0.$$

[4] See also Strauss and Longcope, JCP, **147**, 1998 for a formulation without resistive dissipation.

Time Discretization

Crank-Nicolson semi-discretization in time leads to

$$\begin{aligned}
(J^{n+1} - J^n)/\Delta t + [\mathbf{u} \cdot \nabla J]^{n+\theta} - \eta \Delta J^{n+\theta} &= [\mathbf{B} \cdot \nabla \omega]^{n+\theta} + \{\Phi, \Psi\}^{n+\theta} \\
(\omega^{n+1} - \omega^n)/\Delta t + [\mathbf{u} \cdot \nabla \omega]^{n+\theta} - \nu \Delta \omega^{n+\theta} &= [\mathbf{B} \cdot \nabla J]^{n+\theta} \\
\Delta \Phi^{n+\theta} &= \omega^{n+\theta} \\
\Delta \Psi^{n+\theta} &= J^{n+\theta}
\end{aligned}$$

where $n + \theta$ quantities are calculated as $\xi^{n+\theta} = (1 - \theta)\xi^n + \theta \xi^{n+1}$.

We use PETSc's Jacobian-free Newton-Krylov (JFNK) solver to advance the solution in time.

Inexact Newton Methods

Let $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ and consider solving $F(x) = 0$.

The k^{th} step of classical Newton's method requires solution of the Newton equations:

$$F'(x_k)s_k = -F(x_k).$$

With *inexact Newton methods*, we only require

$$\|F(x_k) + F'(x_k)s_k\| \leq \eta_k \|F(x_k)\|, \qquad \eta_k > 0.$$

This can be done with any iterative method.

Krylov subspace methods only need Jacobian-vector products, which can be approximated by

$$F'(x_k)v \approx \frac{F(x_k + \varepsilon v) - F(x_k)}{\varepsilon}, \qquad \varepsilon \approx \mathcal{O}(\sqrt{\epsilon_{\text{mach}}}).$$

The resulting *Jacobian-free Newton-Krylov* method is easier to use because only function evaluation and preconditioning setup/apply are needed.

Linear Solves

JFNK allows us to focus on solving systems of linear equations

$$\begin{pmatrix} \mathcal{L}_\eta & -\theta \mathbf{B_0} \cdot \nabla & U_{J,\psi} & U_{J,\phi} \\ -\theta \mathbf{B_0} \cdot \nabla & \mathcal{L}_\nu & U_{\omega,\psi} & U_{\omega,\phi} \\ I & 0 & -\Delta & 0 \\ 0 & I & 0 & -\Delta \end{pmatrix} \begin{pmatrix} \delta J \\ \delta\omega \\ \delta\psi \\ \delta\phi \end{pmatrix} = \begin{pmatrix} r_J \\ r_\omega \\ r_\psi \\ r_\phi \end{pmatrix}$$

where

$$\begin{aligned} \mathcal{L}_\eta &= \frac{I}{\Delta t} + \theta(\mathbf{u}_0 \cdot \nabla - \eta\Delta), \\ \mathcal{L}_\nu &= \frac{I}{\Delta t} + \theta(\mathbf{u}_0 \cdot \nabla - \nu\Delta). \end{aligned}$$

Physics-based Preconditioning

This is done by first eliminating δJ and $\delta\omega$, and introducing some approximations[5] to obtain

$$\mathcal{P}\begin{pmatrix}\delta\Psi\\ \delta\Phi\end{pmatrix} \approx \mathbf{\Delta}^{-1}\left[\begin{pmatrix} r_J \\ r_\omega\end{pmatrix} - \mathcal{P}\begin{pmatrix} r_\Psi \\ r_\Phi\end{pmatrix}\right].$$

where

$$\mathcal{P} \equiv \begin{pmatrix} \mathcal{L}_\eta & -\theta\mathbf{B}_0\cdot\nabla \\ -\theta\mathbf{B}_0\cdot\nabla & \mathcal{L}_\nu \end{pmatrix}$$

We then recover the current and vorticity by solving

$$\mathcal{P}\begin{pmatrix}\delta J\\ \delta\omega\end{pmatrix} = \begin{pmatrix} r_J - \theta(\delta\mathbf{u}\cdot\nabla J_0 - \delta\mathbf{B}\cdot\nabla\omega_0 - \{\delta\Phi,\Psi_0\} - \{\Phi_0,\delta\Psi\}) \\ r_\omega - \theta(\delta\mathbf{u}\cdot\nabla\omega_0 - \delta\mathbf{B}\cdot\nabla J_0)\end{pmatrix}.$$

[5] For details see Chacón, Knoll and Finn, JCP, **178**, 2002

Solution Procedure

Systems of equations involving $\mathcal{P}$ are solved with a few iterations of the stationary method obtained from the splitting

$$\mathcal{P} = \mathcal{M} - \mathcal{N}, \qquad \mathcal{M} = \begin{pmatrix} \mathcal{L}_\eta & -\theta \mathbf{B}_0 \cdot \nabla \\ -\theta \mathbf{B}_0 \cdot \nabla & \mathcal{D}_\nu \end{pmatrix}$$

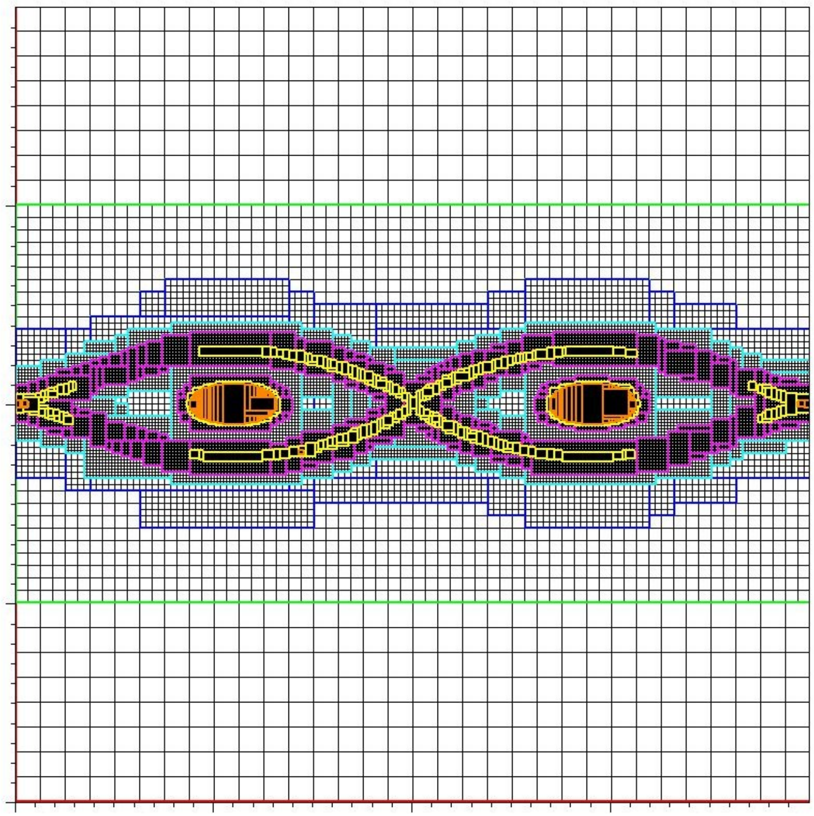
To solve systems of equations involving $\mathcal{M}$, we use the block factorization

$$\mathcal{M} = \begin{pmatrix} \mathbb{I} & -\theta \mathbf{B}_0 \cdot \nabla \mathcal{D}_\nu^{-1} \\ 0 & \mathbb{I} \end{pmatrix} \begin{pmatrix} \mathcal{L}_\eta - \theta^2 \nabla \cdot \mathbf{B}_0 \mathcal{D}_\nu^{-1} \mathbf{B}_0^\mathsf{T} \nabla & 0 \\ -\theta \mathbf{B}_0 \cdot \nabla & \mathcal{D}_\nu \end{pmatrix}.$$

Without spatial adaptivity, the required solves are performed using a multigrid method. With spatial adaptivity, the solves are performed using an AMR-aware variation of multigrid.

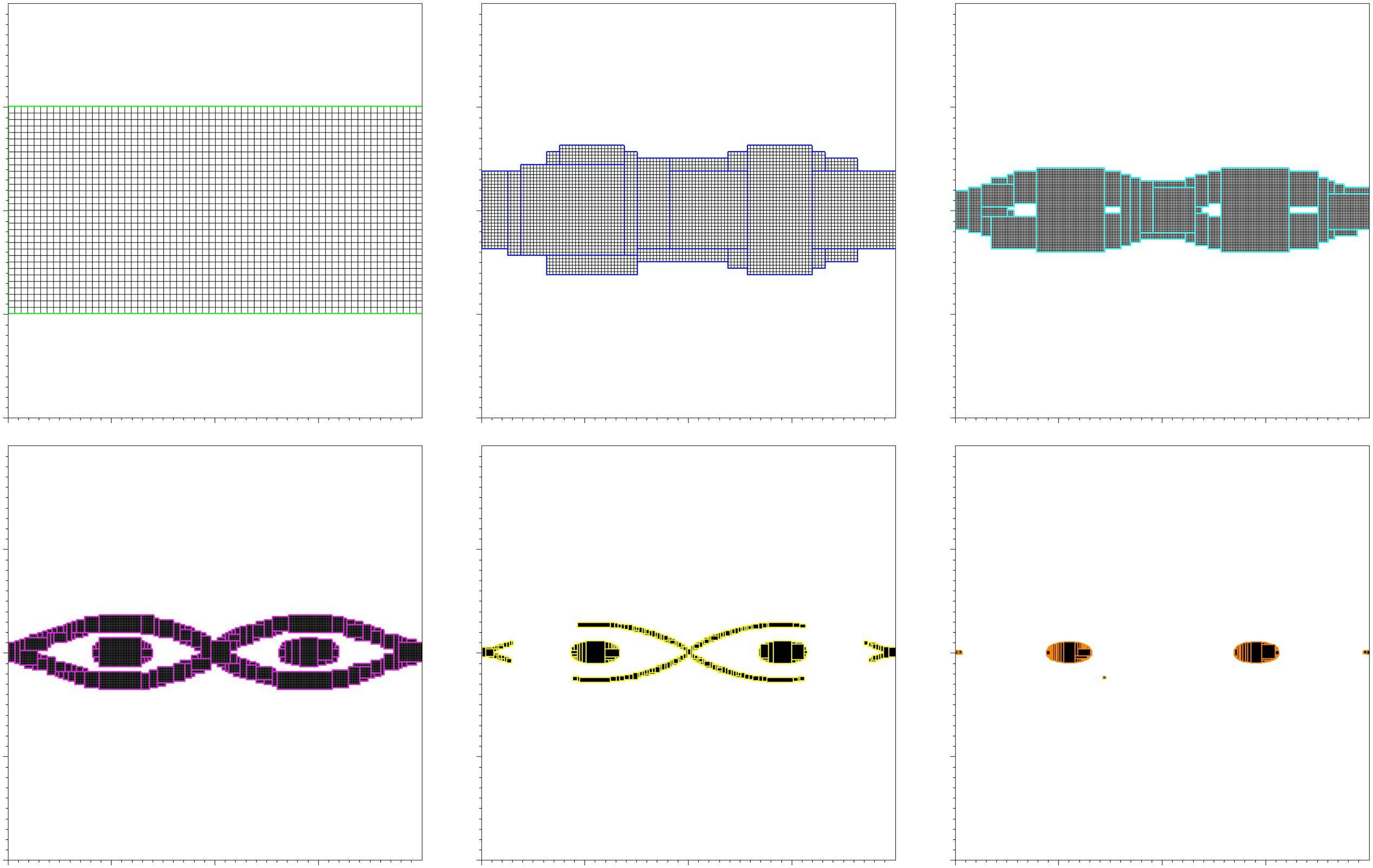
Structured Adaptive Mesh Refinement

Structured adaptive mesh refinement (SAMR) represents a locally refined mesh as a union of logically rectangular meshes.



- The mesh is organized as a hierarchy of nested *refinement levels*.
- Each refinement level defines a region of uniform resolution.
- Each refinement level is the union of logically rectangular *patches*.

Hierarchical Structure of SAMR Grids



Fast Adaptive Composite Grid Method[6]

<u>Algorithm</u>: FAC for $\mathcal{L}x = f$

Initialize: $r^{\underline{h}} = f^{\underline{h}} - \mathcal{L}^{\underline{h}} x^{\underline{h}}$; $f^{h_{L-1}} = I^{h_{L-1}}_{\underline{h}_{L-1}} r^{\underline{h}}$

For $k = L-1, \ldots 1$ {

  Solve/smooth $\mathcal{L}^{h_k} e^{h_k} = f^{h_k}$.

  Correct $x^{\underline{h}_k} = x^{\underline{h}_k} + I^{\underline{h}_k}_{h_k} e^{h_k}$.

  Update $r^{\underline{h}_k} = f^{\underline{h}_k} - \mathcal{L}^{\underline{h}_k} x^{\underline{h}_k}$.

  Set $f^{h_{k-1}} = I^{h_{k-1}}_{\underline{h}_k} r^{\underline{h}_k}$.

}

Solve $\mathcal{L}^{h_0} e^{h_0} = f^{h_0}$.

For $k = 1, \ldots, L-1$ {

  Correct $x^{\underline{h}_k} = x^{\underline{h}_k} + I^{\underline{h}_k}_{h_{k-1}} e^{h_{k-1}}$.

  Update $r^{\underline{h}_k} = f^{\underline{h}_k} - \mathcal{L}^{\underline{h}_k} x^{\underline{h}_k}$.

  Set $f^{h_k} = I^{h_k}_{\underline{h}_k} r^{\underline{h}_k}$.

  Solve/smooth $\mathcal{L}^{h_k} e^{h_k} = f^{h_k}$.

  Correct $x^{\underline{h}_k} = x^{\underline{h}_k} + I^{\underline{h}_k}_{h_k} e^{h_k}$.

}

[6] McCormick and Thomas, Math. Comp., **46**, (1986).

Tearing Mode Problem

Initial conditions:

$$\begin{aligned}\Psi_0(x,y) &= \tfrac{1}{\lambda}\ln[\cosh(\lambda(y-\tfrac{1}{2}))] \\ \Phi_0(x,y) &= 0 \\ \omega_0(x,y) &= 0\end{aligned}$$

Boundary Conditions:

Periodic in x and homogenous Dirichlet in y.

Perturbation:

$$\delta\Psi = 10^{-3}\cos(\tfrac{\pi}{2}x)\sin(\pi y)$$

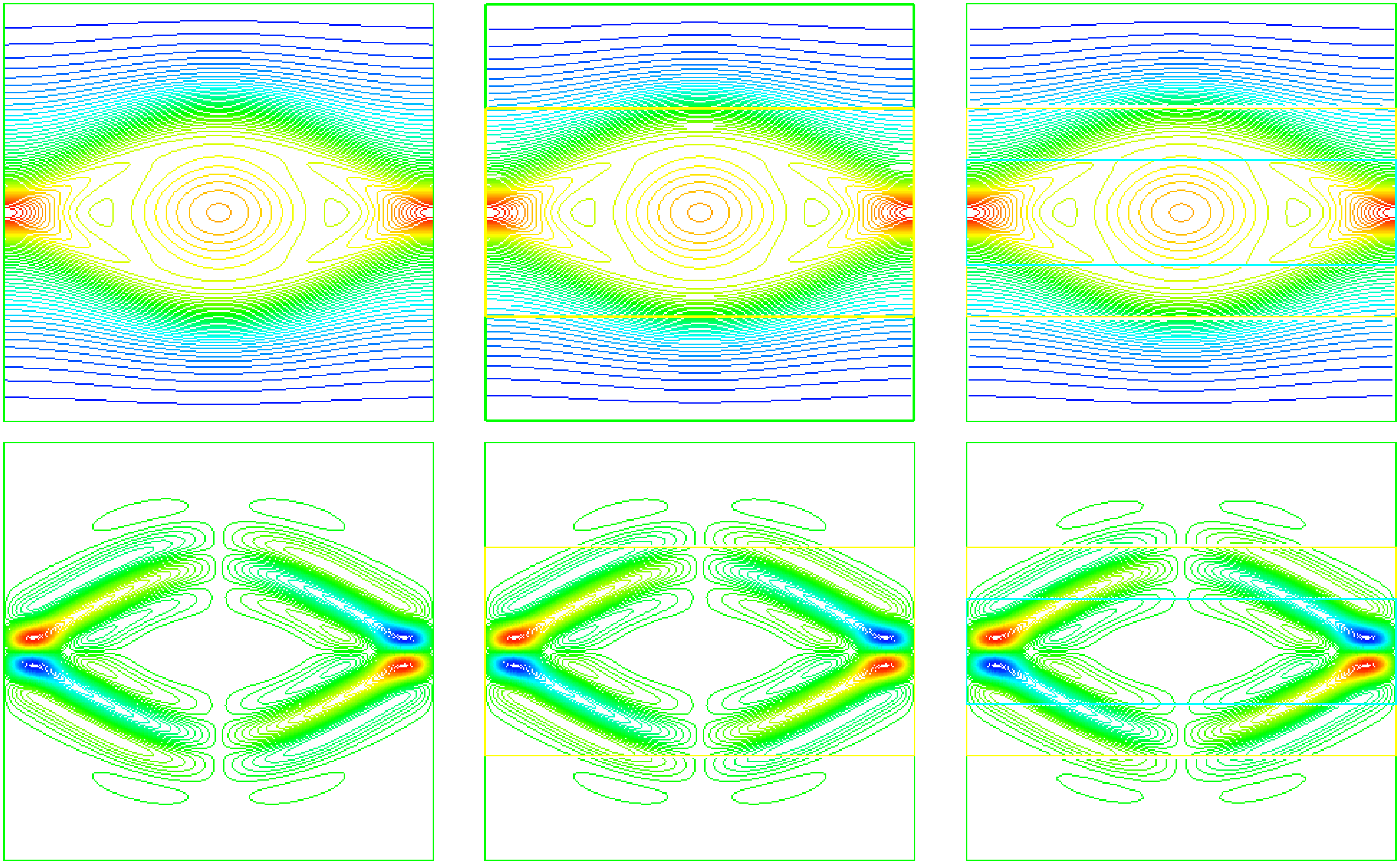
Parameters:

$\Omega = [0,4]\times[0,1]$, $\lambda = 5$, $\eta = \nu = 10^{-3}$

Refinement criteria:

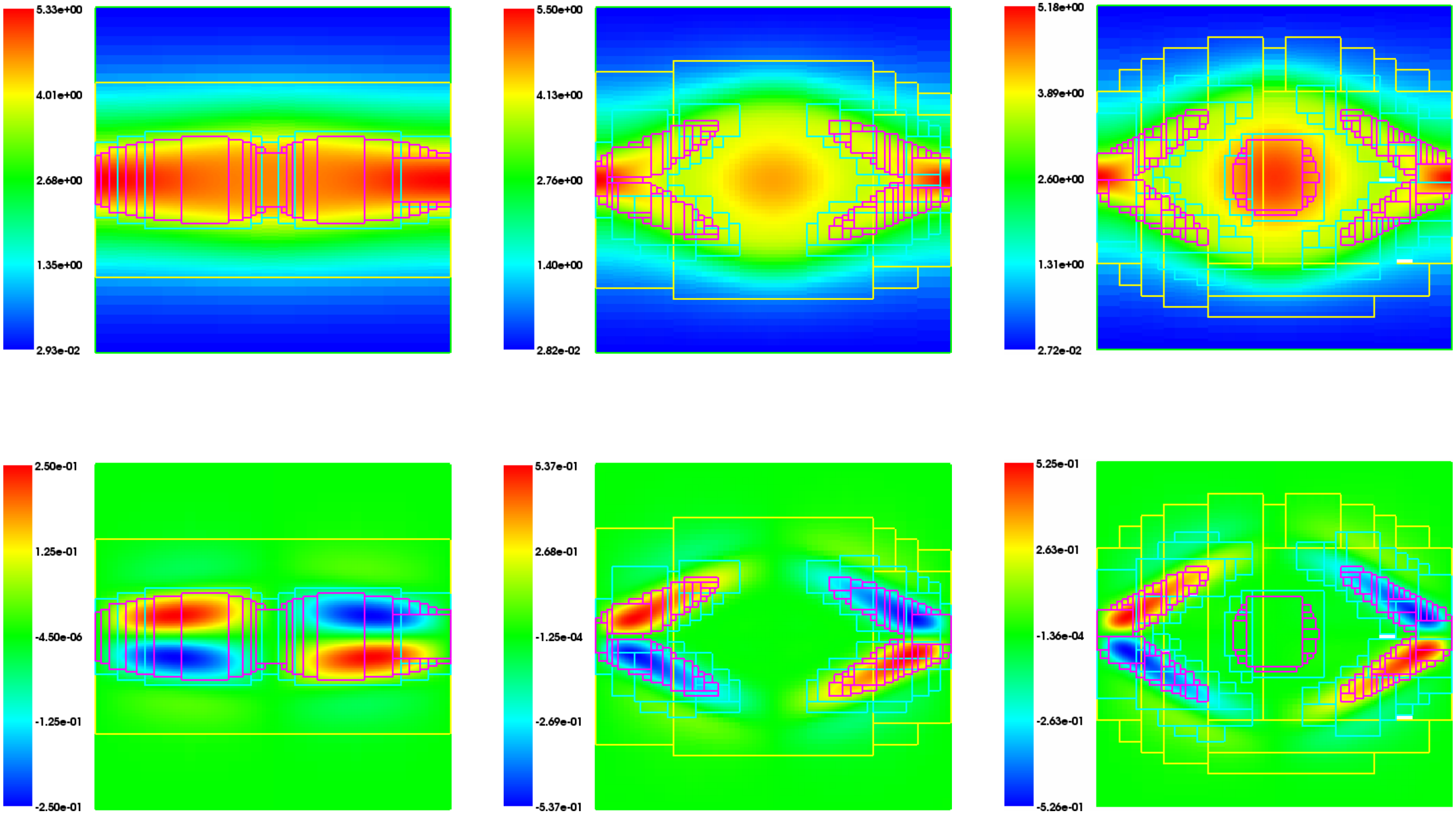
- magnitude of J
- curvature of ω

Tearing Mode Comparison at $t = 120$



256B+0 128B+1 64B+2

Tearing Mode Results



$t = 50$ $\qquad$ $t = 120$ $\qquad$ $t = 200$

Tearing Mode Performance

	NNI					NLI				
Levels	1	2	3	4	5	1	2	3	4	5
32×32	1.5	2.0	2.0	2.1	2.5	3.4	7.9	12.0	19.3	33.7
64×64	1.8	2.0	2.0	2.4	–	6.5	11.7	19.1	33.2	–
128×128	1.8	2.0	2.4	–	–	12.5	20.1	27.2	–	–
256×256	1.9	2.0	–	–	–	19.9	27.5	–	–	–
512×512	1.9	–	–	–	–	26.3	–	–	–	–

$\eta_k = 0.1$, $\epsilon_{rel} = \epsilon_{abs} = 10^{-7}$, 2 SI iterations, V(3,3) cycles

Island Coalescence

Initial conditions:

$$\begin{aligned}\Psi_0(x,y) &= -\lambda\ln[\cosh(\tfrac{y}{\lambda})+\epsilon\cos(\tfrac{x}{\lambda})]\\ \Phi_0(x,y) &= 0\\ \omega_0(x,y) &= 0\end{aligned}$$

Boundary Conditions:

Periodic in x and homogenous Dirichlet in y.

Perturbation:

$$\delta\Psi = 10^{-3}\cos(\pi x)\cos(\tfrac{\pi}{2}y)$$

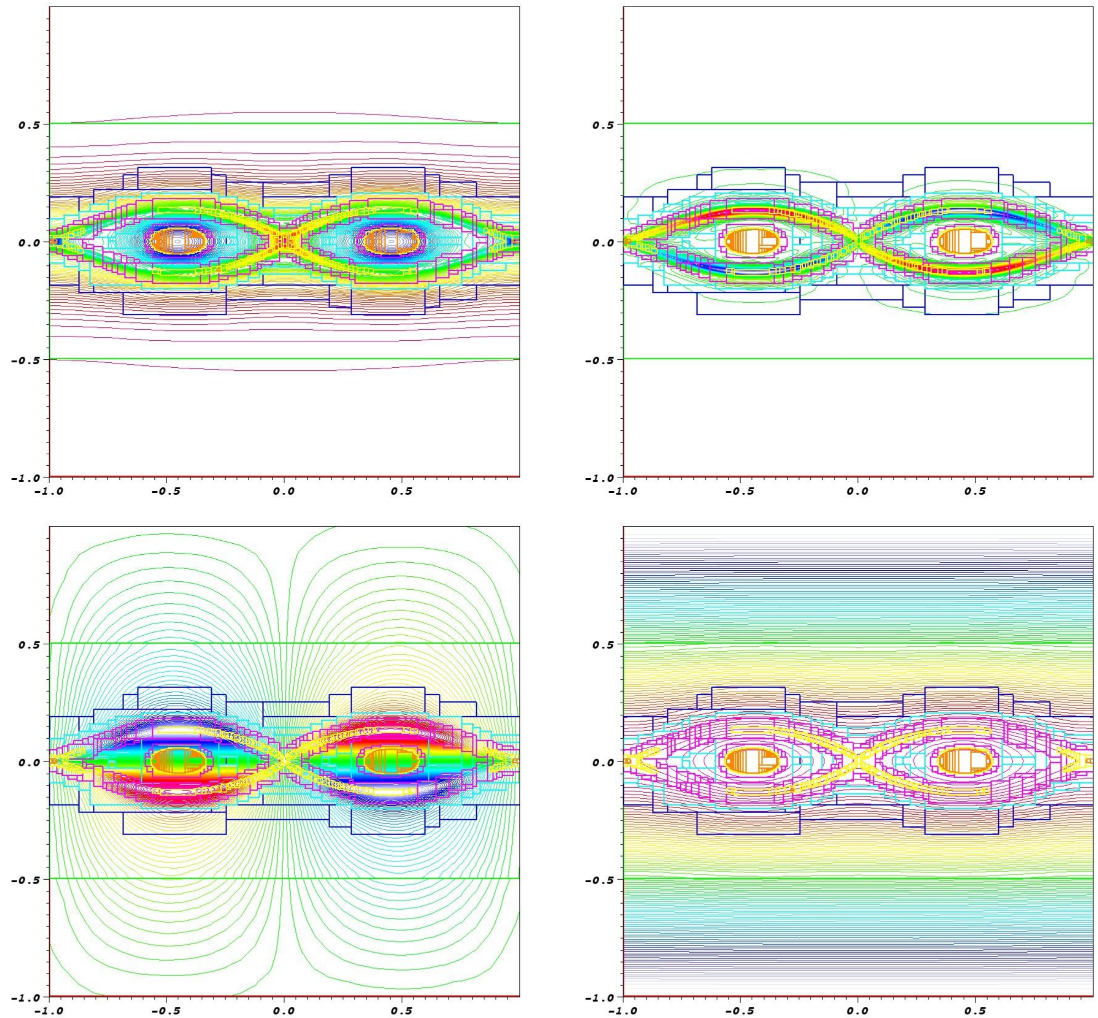
Parameters:

$$\Omega = [-1,1]\times[-1,1],\ \lambda = \tfrac{1}{2\pi},\ \epsilon = 0.2,\ \eta = \nu = 10^{-4}$$

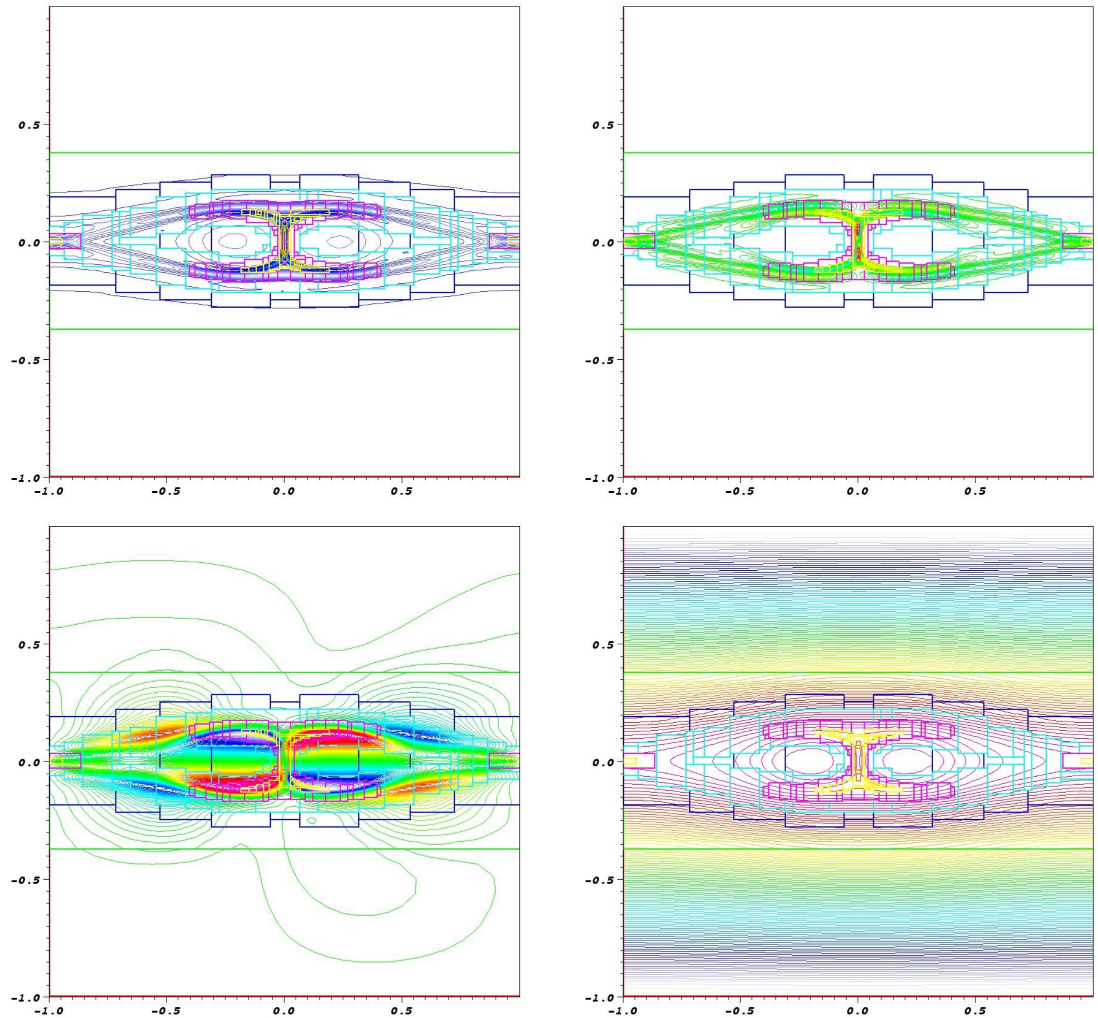
Refinement criteria:

- magnitude, curvature of J
- curvature of ω

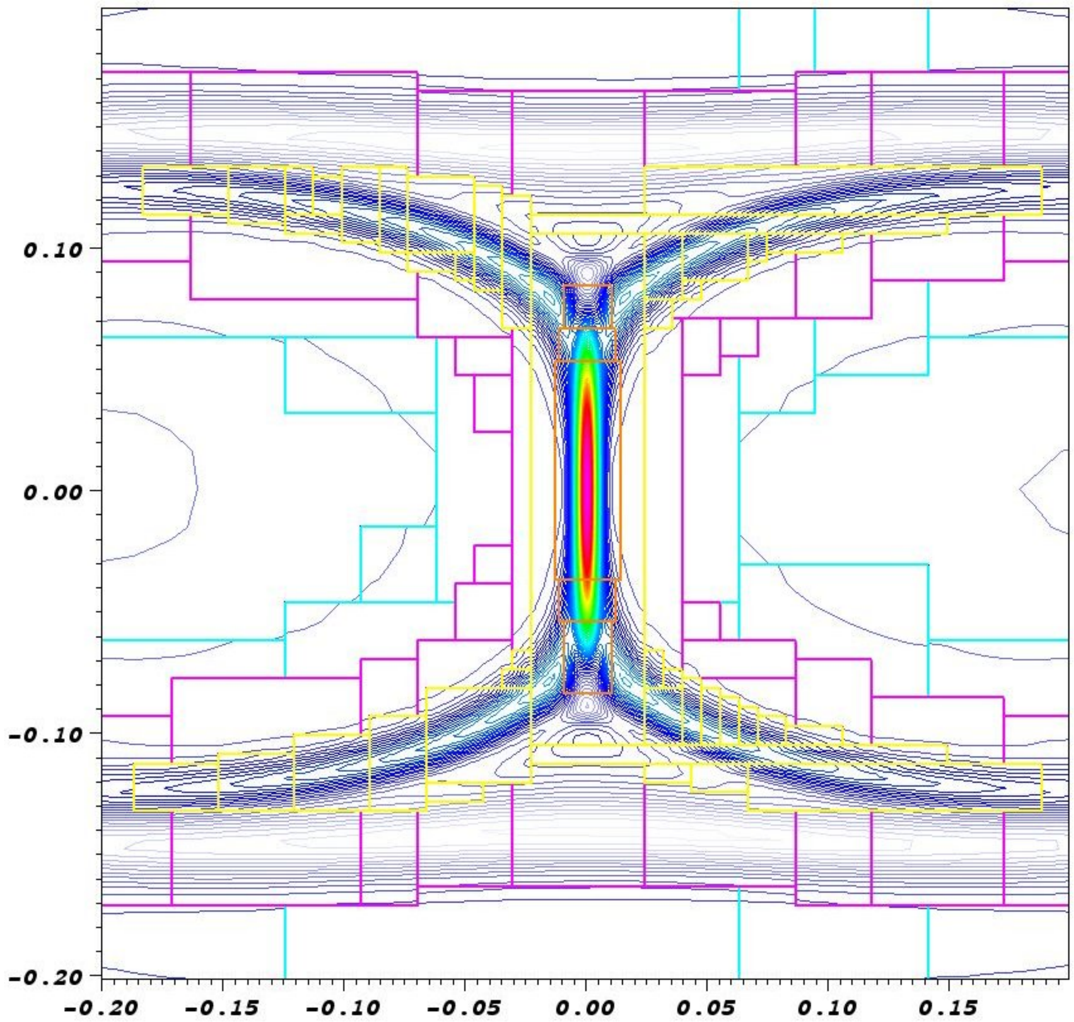
Island Coalescence Results at $t = 4$



Island Coalescence Results at $t = 8$



Island Coalescence Current Sheet Detail



Island Coalescence: Performance

Final time:

$t = 20$

Time increment:

$\Delta t = 0.01$

JFNK parameters:

$\eta_k = 0.05$, $\epsilon_{rel} = \epsilon_{abs} = 10^{-6}$

SI Preconditioner parameters:

2 iterations, V(3,3) cycles

Newton iterations per timestep:

3.9

Linear iterations per timestep:

9.0

Tilt Instability

Initial conditions:

$$
\begin{aligned}
\Psi_0(x,y) &= \begin{cases} \frac{2}{kJ_0(k)} J_1(kr)\cos(\theta) & \text{if } r \le 1 \\ (r - \frac{1}{r})\cos(\theta) & \text{if } r > 1 \end{cases} \\
\Phi_0(x,y) &= 0 \\
\omega_0(x,y) &= 0
\end{aligned}
$$

Boundary Conditions:

Periodic in x and homogenous Dirichlet in y.

Perturbation:

$$\delta\Phi = 10^{-3} e^{-r^2}$$

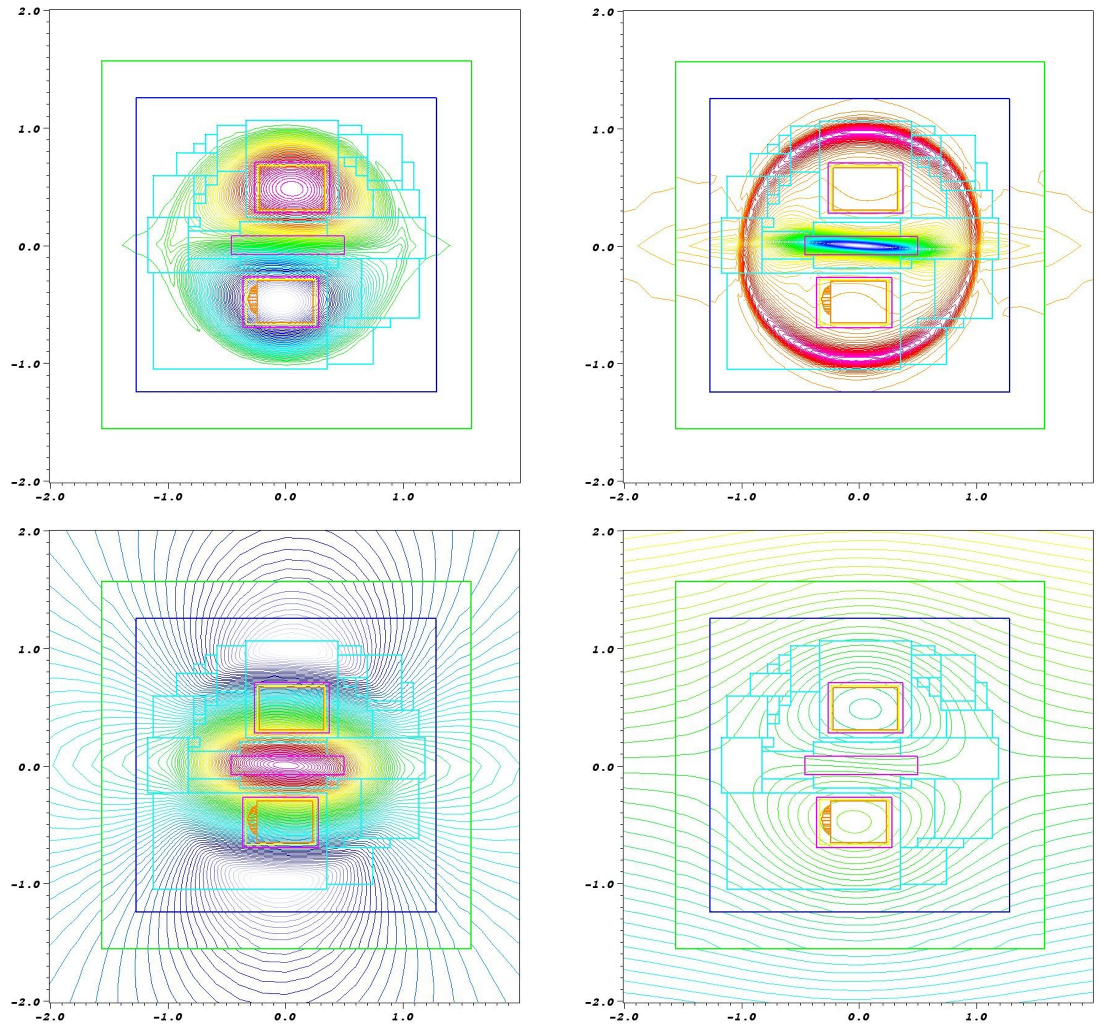
Parameters:

$$\Omega = [-2\pi, 2\pi] \times [-5, 5],\ J_1(k) = 0,\ \eta = \nu = 10^{-3}$$

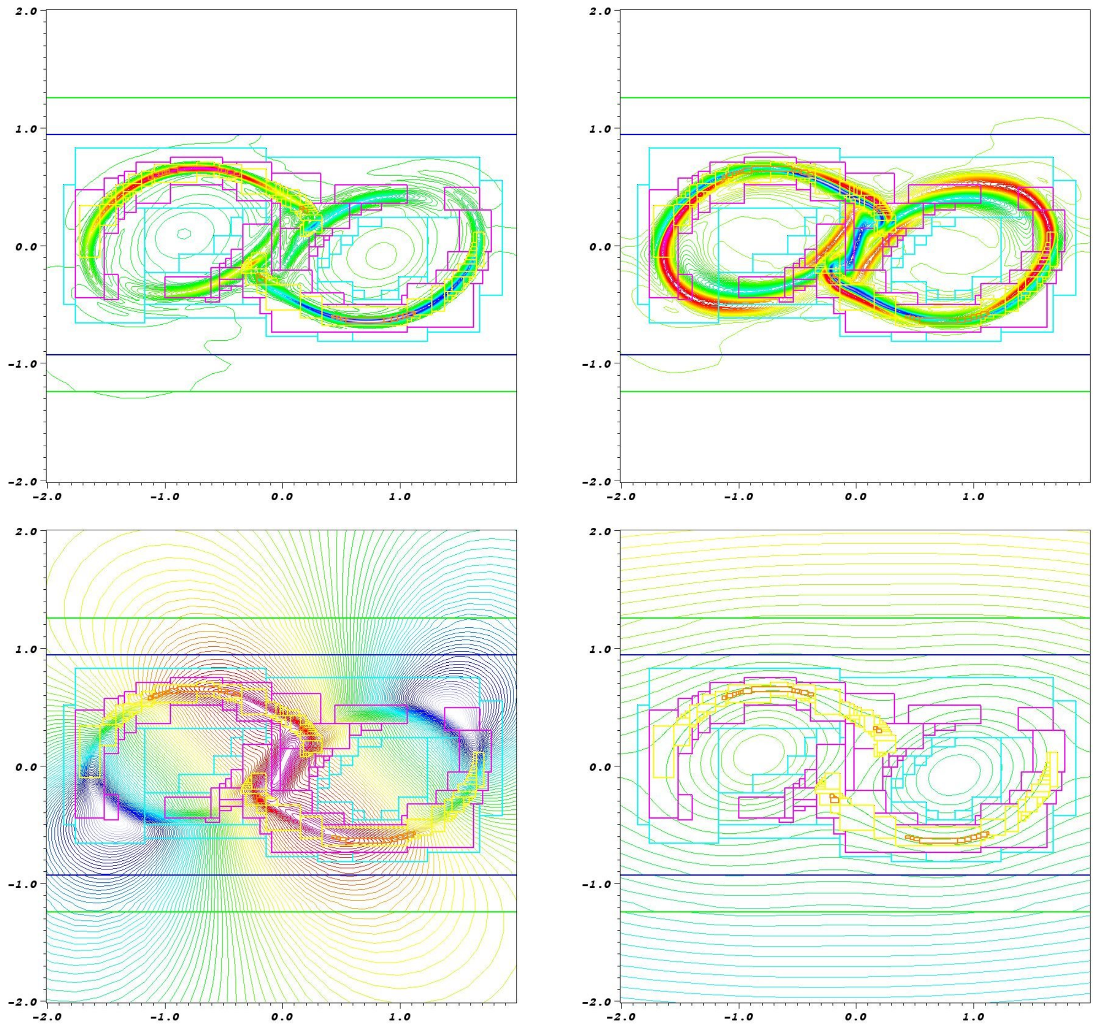
Refinement criteria:

- magnitude, curvature of J
- curvature of ω

Tilt Instability Results at $t = 4$



Tilt Instability Results at $t = 7$



Tilt Instability: Performance

Final time:

$t = 20$

Time increment:

$\Delta t = 0.005$

JFNK parameters:

$\eta_k = 0.01$, $\epsilon_{rel} = \epsilon_{abs} = 10^{-6}$

SI Preconditioner parameters:

1 iteration, V(3,3) cycles

Newton iterations per timestep:

3.7

Linear iterations per timestep:

17.6

Conclusions

- Spatial adaptivity allows us to efficiently resolve fine scale features.
- Multilevel preconditioning strategy controls work required for the implicit solves and makes implicit integration competetive.

Future Work

- Verify correctness of locally refined calculations.
- Quantify the impact of local mesh refinement.
  - Problem size.
  - Execution time.
- Determine scaling of linear iteration counts with amount of local refinement for island coalescence and tilt instability problems.
- Determine performance for "interesting" values of η and ν.
- Enhance parallel performance.
- Local time step error control (under development).